Journal of Artificial Intelligence and Capsule Networks (ISSN: 2582-2012)
www.irojournals.com/aicn/

Eye Strain Expression Classification using Attention Capsule Network for Adapting Screen Vision

Chiranjibi Pandey[1], Sanjeeb Prasad Panday[2]

[1]Department of Electronics and Computer Engineering, IOE Pulchowk Campus, Tribhuvan University, Kathmandu, Nepal

[2]Department of Electronics and Computer Engineering, IOE Pulchowk Campus, Tribhuvan University, Kathmandu, Nepal

Email: [1]078msice006.chiranjibi@pcampus.edu.np, [2]sanjeeb@ioe.edu.np

Abstract

Beside the conventional facial expression recognition methods, the research focuses on developing a system for recognizing various eye expressions under different screen conditions. This research deals with the use of Capsule Network (a recent Deep Learning algorithm) to enhance facial expression recognition capabilities and to develop adaptive screen technologies aimed at mitigating digital eye strain. The main objective of this research is to engineer a sophisticated system that employs the capabilities of Capsule Nets to recognize the various expressions that user makes and based on the recognized expression, dynamically modify screen settings, ensuring optimal user visual comfort.

The research primarily concentrates on the exploration and application of various Capsule Net architectures designed for the recognition of expressions related to eye strain. The baseline model utilized elementary convolutional layers which feed into subsequent fully connected layers for the task of classification. The model has since been refined by incorporating advanced techniques such as attention mechanisms and more sophisticated network architectures where the classification is done by Capsule Network. Results have demonstrated a modest enhancement in the Capsule Net's predictive performance, attributed

DOI: https://doi.org/10.36548/jaicn.2024.2.005
Received: 04.04.2024, received in revised form: 21.04.2024, accepted: 02.05.2024, published: 18.05.2024

to its superior spatial and hierarchical processing of facial features, in comparison to conventional deep learning approaches. The final model has an accuracy of 82.27%. As a final system the model has been deployed to an application to process frames from video camera in the device and make prediction to prompt the notifications or recommendations.

Keywords: Capsule network, Eye Expressions, Attention Mechanism, Deep Learning Algorithms

1. Introduction

Growing use of digital devices with high screen exposure has come out to be a significant problem in modern society. This problem can be mitigated or reduced to some extent by using the computer vision approach by automating the detection of visual discomforts that user experiences. Hence, improved and reliable expression recognition method can be incorporated in this field.

CNNs have shown significant results in facial expression recognition purpose, achieving high accuracy by using large datasets and fine-tuning pre-trained models. However, challenges remain in handling variations in facial expressions, occlusions, and pose changes. Some shortcomings of CNN that can be addressed by CapsNet are viewpoint variations, spatial hierarchies, deformation and non-linear transformations, interpretability and non-linearity and robustness to occlusions.

1.1 Objectives

The main objectives of this study are as follows:

- To explore the potential of capsule networks in improving expression recognition accuracy specific to eye expressions.
- To develop an intelligent system that automates screen adaptation techniques for mitigating digital eye strain based on capsule network eye strain expression recognition.

2. Related Work

Use of various CNN models for facial expression recognition regarding digital eye strain has been discussed by (Mutanu 2023) and the hierarchical representation of facial images for capsule network is explained by (S. Sabour 2017) . For the study relating to the ophthalmology, study was done in (L. O'Hare 2013) to find out the root causes of the eye discomfort and ultimately causing the eye expressions. These bases are superficially taken for the foundation of the study in this research. The theoretical basis has been used for the classification of eye strain expressions into several classes. The root cause regarding digital eye strain has been pointed out and used in the research. For the purpose of recognizing facial expressions in unrestricted situations, numerous strategies have been designed. For instance, researchers in (C. Shan 2009) presented a use of Local Binary Pattern (LBP) to recognize the expressions and in (Y.Hu 2008) present the use of multi-view approach for the recognition task. Similarly biometric learning method has been proposed by (Z. Liu 2016) where the geometric calculations are used to deduce the decisions on expression recognition. One to one feature and expression mapping has been explained in (Boosting–poof: Boosting part based one vs one feature for facial expression recognition 2017). Use of Ensemble technique in CNN has been put forward by (Y. Fan 2018) making use of the multiple models for the prediction and integrating the decisions using bagging or voting methods. To improve the feature concentration in the learning attention net is put forward in the (D. Marrero 2019). Use of conventional machine learning approaches like SVM and k-NN classification has been put forward by (Y. Yan 2020). In recent years, the substantial enhancement in chip processing capabilities alongside innovatively designed network structures has led researchers across various disciplines to pivot towards deep learning methodologies. Similarly, in the field of Facial Expression Recognition (FER), deep learning approaches have set new benchmarks. Sun and colleagues in (Y. Sun 2013) devised a novel deep convolutional network inspired by silicon structures, incorporating an innovative attribute propagation technique to bridge the gap between diverse sources, facilitating the learning of facial expressions through multitasking to infer interpersonal relations. Tran and his team in (D. Tran 2015) introduced an advanced C3D model employing 3D convolution over extensive supervised training datasets to capture spatiotemporal characteristics.

Several studies have leveraged this model for FER tasks involving sequential images. Zhang and associates in (K. Zhang 2017) developed a part-based hierarchical bidirectional recurrent neural network for the temporal analysis of facial expressions. Liu and his team (ZT. Liu 2019), for instance, utilized class-pairwise Mid-Level descriptors for localized regions to extract Mid-Level features and employed Adaboost for the selection of the most distinctive features. Furthermore, the significance of shape and texture in recognizing facial expressions has been explored, with Peng and Yin (Y. Peng 2019) proposing an appearance-based facial expression synthesis framework. Barman and Dutta (A. Barman 2019) explored how shape and texture attributes impact FER. However, under unconstrained conditions, these models show less robustness and all these methods based on CNN and other traditional methods for FER do not consider both the characteristics of face structure and the information of feature spatial relation. Also, they neglect the spatial relationship between facial action units (AUs). To solve these problems, we propose eye AU aware-Capsule Network with deep feature representation using eye AU-aware attention maps. The property of capsule network needs the features to be directional, and can capture relative relationships between different landmarks and AUs. The baseline paper for the research is (Mutanu 2023) which has use simple stacked-CNN for the expression classifications in the FER dataset. The research is based on the improvement of the performance of the method in the paper by using CapsNet for the eye expression classification.

3. Methodology

The methodology of the research, mainly comprises of dataset preparation for the eye strain images and their labels, their pre-processing, development of the classification models, training of the developed models, using the trained models for the inference, analysing, validation, and finally using the final model for the making of automated screen parameter adjusting.

The proposed method of the research is explained by the Figure 1 a) and Figure 1 b) as:

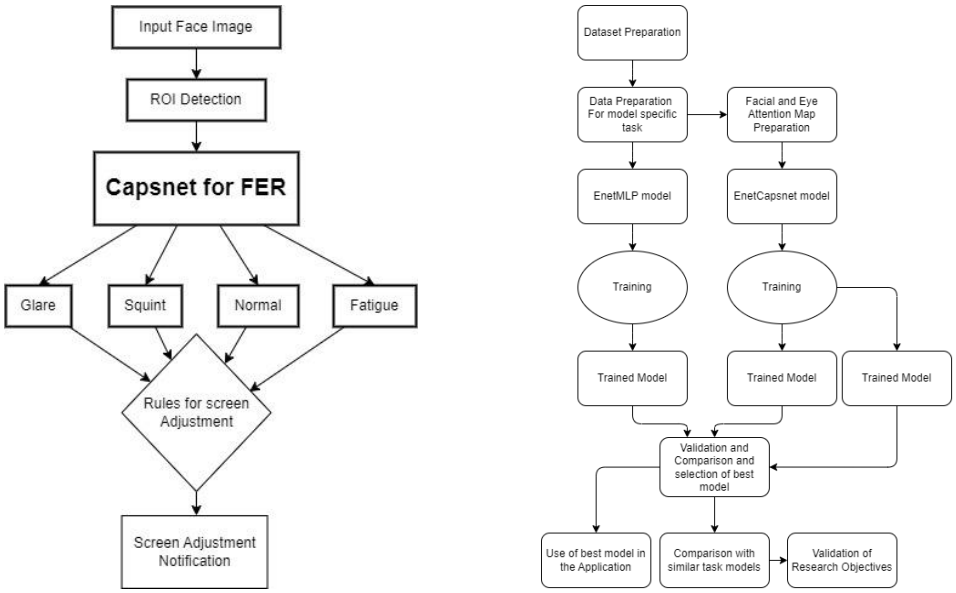
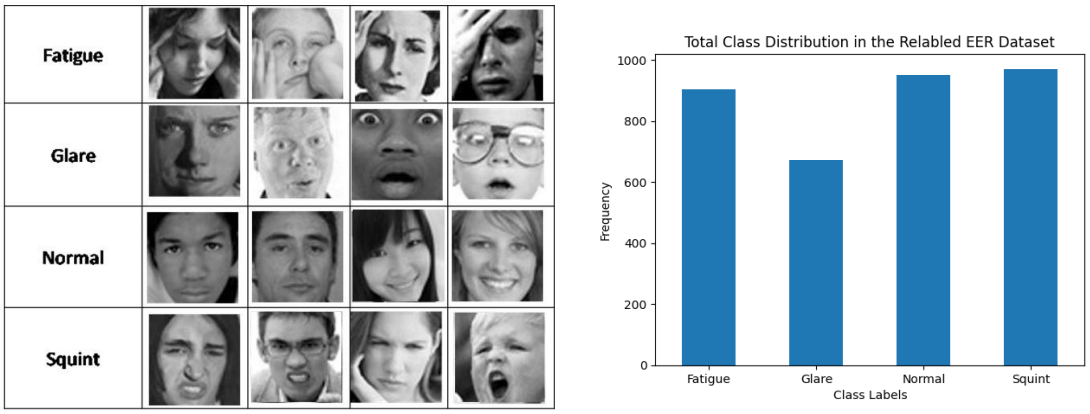


Figure 1. a) Overall Block Diagram of the Proposed System **b)** Methodology of the Research

3.1. Data Source

For this research, facial images from the FER datasets were manually reclassified to reflect expressions associated with digital eye strain. From the FER2013 CK++48, and RAF-DB open datasets, a total of 3500 images were sampled which are best indicating the eye expressions and are reclassified from traditional expressions of emotions (such as happiness, sadness, anger, or fear) to expressions based on digital eye strain (namely squint, glare, and fatigue). In the reclassification process, the Squint category was assigned to images showing inward eyebrows and squinted eyes, while the Glare category included images with raised eyebrows and wide-open eyes. Images depicting closed eyes, faces covered or held with hands, or yawning were designated as Fatigue. Images not displaying any of these specific expressions were placed into the Normal category. Expressions not demonstrating squint, glare, or fatigue were deemed normal. The snapshots and the class distribution are shown in Figure 2.



a) **b)**

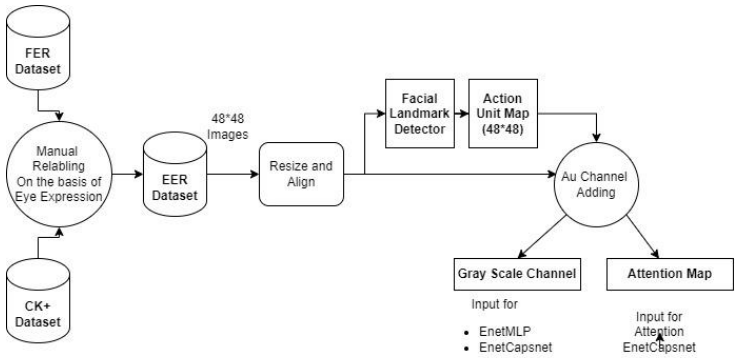
Figure 2. a) EER Dataset Snapshot b) Classes Distribution of the Prepared Dataset

3.2. Data Preprocessing

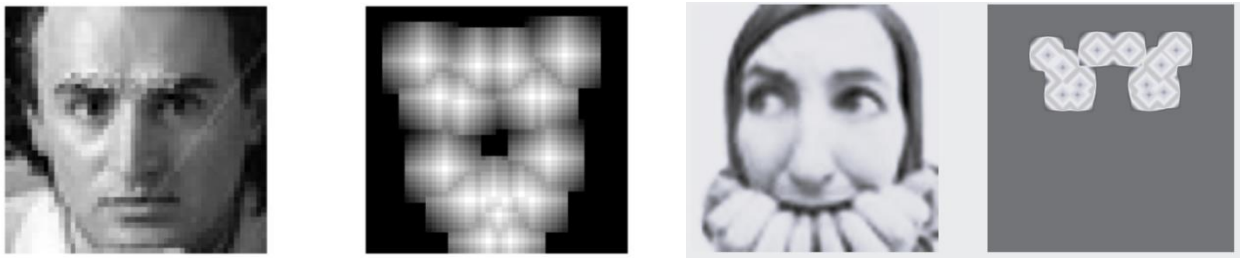
After the datasets are collected the preliminary process of reclassification of images based on the eye strain expressions [Fatigue, Glare, Normal, Squint] was carried out. From this the final dataset has been built that has around 3500 images and is named as EER dataset as shown in Figure 3. The obtained pool of images is then converted to consistent size and dimension. For the research, conversion involved various data requirements. In the initial phase 48*48 images were used for the research but this experiment couldn't give the result to the mark and later the more consistent dataset was prepared by converting every images to RGB image of size 100*100. Conversion to RGB image was done because of the diversity of image types obtained from initial 3 different datasets. This resizing ensures consistency and is followed by an alignment step. The proper alignment of the facial structure is important for the face and landmark detection for attention integration and reliable feature extraction in further stages, which can significantly impact the performance of the expression recognition models.

After resizing and alignment, a Facial Landmark Detector is employed from the dlib library. This landmark detector identifies 68 key facial points, which are vital for recognizing and integrating the structure and subtle changes in facial expressions. The detected landmarks are then used for the generation of an Action

Unit (AU) Map of the same dimensions as the input images (48x48) or (100*100) as required by the model architecture. The action unit map represents the intensity and direction of various facial features it is done by assigning the intensity to the landmarks detected point varying outwards representing the vicinity of the landmark region. For the finalized model for inference and to increase the accuracy, the action unit maps for eye and vicinity region was only created. The AU map is then combined with the original images to get the two-channel images or four-channel images. This integration augments the original data with detailed expression information to the attention map of the eye region enhancing the model's capacity to recognize specific eye expressions by providing additional contextual boosting clues.



a)



b) **c)**

Figure 3. a) Pre-processing Pipeline **b)** Image and Facial Attention Map **c)** Image and Eye Attention Map

3.3. Enet with Capsule Network for Classification

The Enet (Green enclosed Architecture) is our feature extraction module which begins with a convolutional layer applying various filters (64 to 512) to the image input including batch normalization and ReLU activation. Each sequence ends with pooling operations to reduce the dimensions and effectively capture the hierarchy of features which are fed to the classifier.

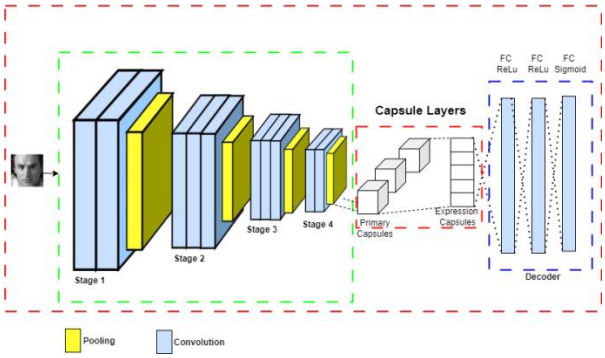


Figure 4. EnetCapsnet

Ablating the MLP (Multilayer-Perceptron) from the EnetMLP and introducing capsule layers capsule layers for classification instead of MLP gives our EnetCapsNet model without any attention mechanism. Capsule Network is the neural network of capsules. Capsules are groups of neurons that specialize in detecting and encoding various aspects of the input, such as spatial hierarchies between features. In this architecture, the initial stages are similar to the EnetMLP, utilizing convolutional and pooling layers, and the classification is done with the capsule layers followed by the decoder or reconstruction network which is again a fully connected neural network. This has been incorporated to aid the capsule learning with additional reconstruction loss.

- **Primary Capsules:** After feature extraction, the network applies a capsule layer. The primary capsules layer is the first capsule layer, which does not use routing by agreement. This layer converts the feature maps obtained from ENet into a set of capsule outputs (vectors), where each capsule learns to detect specific features present in the image and encodes into a vector containing additional instantiations parameters along with activation strength.

- **Expression capsules:** The output from the primary capsules is then passed to the five Expression capsules as there are five classes of expressions in EER Dataset. This layer uses a dynamic routing mechanism to allow capsules to learn a part-whole relationship. Each primary capsule tries to predict the output of higher-level expression capsules, and through an iterative process, the network determines how to route information between layers.

- **Decoder:** The decoder is a reconstruction network comprising several fully connected (Linear) layers with ReLU activation functions and a final sigmoid activation. It takes the output from the Expression capsules and attempts to reconstruct the original input image, which can help the network to learn encoding the instantiation parameters of the input image.

- **Forward Pass:** During the forward pass, the input image x is first processed through ENet and then through the primary and expression capsule layers. The vector output from the expression capsules is then squashed, and a softmax function is applied to the length of these vectors to represent the probability of each class. The class with the highest probability is selected as the prediction. If the true label y is provided, a mask is applied to the capsule outputs, ensuring that only the capsule output corresponding to the true label is used for reconstruction. The reconstruction loss is often used alongside the classification loss to train the network, ensuring that the capsules learn robust feature representations. The involved squashing function (1), corresponding margin loss (2) and reconstruction loss (3) for the training are as follows:

$$squash(s_j) = \frac{\|s_j\|^2}{1+\|s_j\|^2}\frac{s_j}{\|s_j\|} \quad (1)$$

$$Margin\ Loss: L_c = T_c.\max(0, m_+ - \|v_c\|)^2 + \lambda(1 - T_c)\max(0, \|v_c\| - m_-)^2 \quad (2)$$

$$Reconstruction\ Loss: L_r = \|X - X'\|^2 \quad (3)$$

$$Total\ Loss: L = L_c + \alpha L_r \quad (4)$$

3.4. Attention Aware CapsNet

In attention capsNet the attention maps provide a way to focus on specific parts of the input image that are more relevant for the task of recognition such as key facial landmarks like the eyes and mouth when detecting expressions. The attention maps modify the feature maps obtained from the convolutional layers, emphasizing important features while diminishing the less relevant ones.

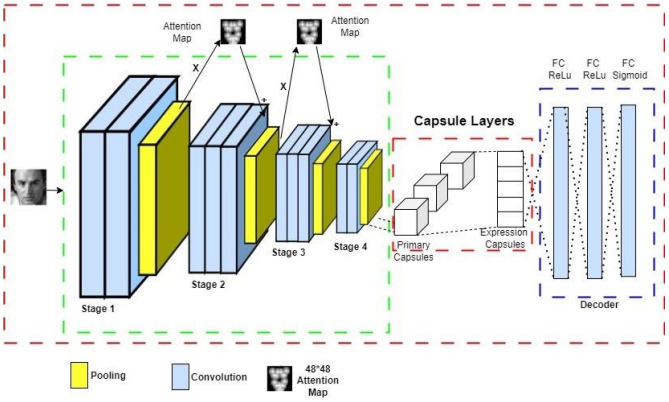


Figure 5. Attention aware Capsnet

This is done by multiplying the aligned output of the stage1 and stage2 output with the corresponding feature maps and then down-sampling to the dimension of the subsequent layer for the addition purpose. This process is done in the separate path from the original extraction flow to integrate additional information to the typical extraction network. This process occurs before feeding the data into the capsule layers, allowing the network to make more informed predictions based on the most important features or the features of interests. Like the EnetCapsnet, this model also includes primary and expression capsules as well as a decoder. The attention mechanism has the potential to significantly enhance the model's performance by mimicking the human visual attention system. The used attention mechanism is applied in two phases of experiment. First is the one involving the overall facial landmarks to create the facial attention map. In the subsequent phase, only the eye region and vicinity was used for the attention map to create the eye attention map.

3.5. Rule Based Screen Parameter Adjusting Application

An eye expression responsive application has been developed using TKinter graphics library. The development flow is shown in the Figure (6). In the application the real time video feed is used from the web camera and is run in a loop until a program close signal is given to the application. In the loop periodic frames are extracted from the video feed and are resized and converted to the format for the display in the application

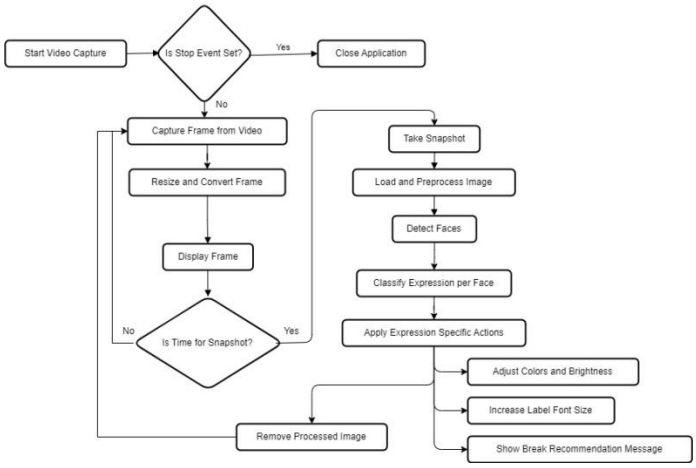


Figure 6 Real-Time Application Workflow of the Application

If a predicted class persists for the threshold time, then the IF-then rule is applied to notify the user asking suitable parameter adjustment. This can be automated too. However for the responsiveness of the application, user prompt dialogue is rendered and the corresponding action is taken like font increment, color changing, closing of the application and so on.

3.6. Tools and Experimental Setup

Table 1. Experimental Tools and Setups

Tools & Frameworks	Usage in the Thesis
Python	Fundamental Language to build the system
Pytorch	Tool to build the architectures and train
Google Colab	Cloud based hardware platform to develop and train models.
Numpy, Pandas	Libraries to handle data loading and data structures
OpenCV	Image Processing and manipulation
dlib	Library for Face ROI extraction
LaTeX	Report Documentation
TensorboardX	Tool to visualize train metrics, images and T-SNE visualization
TKinter	Demonstration Application Design

3.7. Training Analysis

The described models of the study were trained in the configuration above up to the 20 epochs in the dataset of eye expressions. Keeping in mind the computational resources available careful optimization of the process was done in order to get the desired results. From the plots in Figure 7 it is clear that the task of eye expression classification based on eye attention has learnt well compared to the other one.

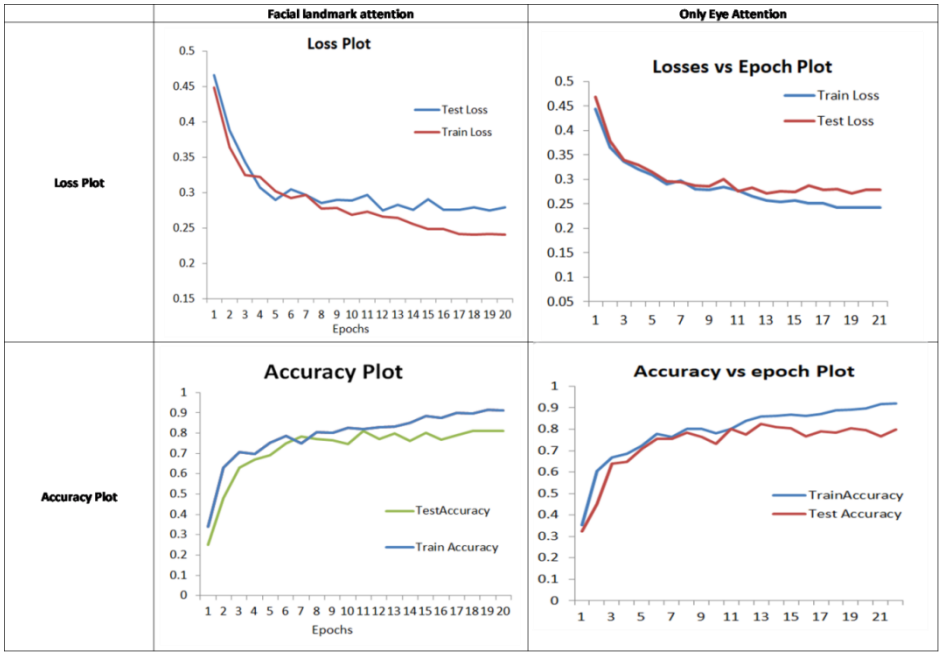


Figure 7. Train and Test Plots of Two Final Models

3.8. Hyper-Parameters Configurations

Table 2. Training Configuration and Hyperparameters Choice

Parameters	EnetMLP	EnetCapsnet	AttentionEnetCapsnet
Batch Size	16	16	16
Optimizer	Adam	Adam	Adam
Learning Rate	lr_scheduler	0.0001	0.0001
Loss Function	Crossentropy	M+0.0005*R	M+0.0005*R
m+ (Positive Margin)	X	0.9	0.9
m- (Negative Margin)	X	0.1	0.1

Primary Capsules	X	8	8
Routing by Agreement	X	Yes	Yes
Routing Iterations	X	3	3
Training Hardware	T4 GPU	V100 GPU (Colab)	A100 GPU (Colab)
M: Margin Loss			
R:Reconstruction Loss			
lr_scheduler: Learing Rate Scheduler			

4. Results and Discussion

4.1. Capsule Activation for Various Test Images

Using the final model, some of the self and general images were preprocessed to extract the face ROI and were fed to the model for the prediction. The activation value of respective class capsules is clearly refined in every routing iterations and finally the primary and the expression capsules agree upon for routing. The iterative improvement of decision making in capsule network is clearly explainable with this visualization. The images in right in Figure 8 are real-time app demos.

Figure 8: Corresponding Eye Expression Capsule Activations, Expression of Eye Strain recognized, Working of the Developed Application

4.2. Test Set Analysis of the Models

The output of the expression capsules was visualized using the t-SNE which shows the significant classified clusters. The plots in Figure 9 suggest that the models sometime struggle in classifying the class between ‘squint’ and ‘fatigue’. This problem is seen more in the eye only attention model as the fatigue images in the dataset involve the hands holding the face. There may have been similarity in eyes orientation in both classes. So using the overall facial landmark in the first model has shown a little better separation between squint and fatigue.

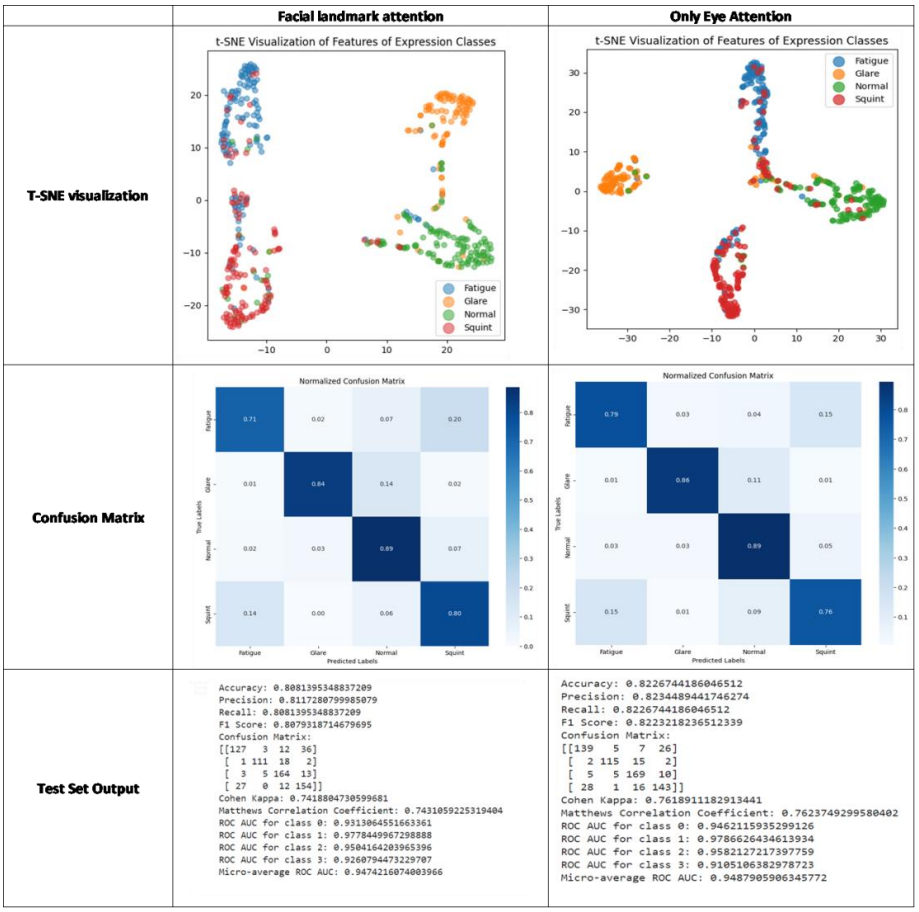


Figure 9. Test Set Results of Two Final Experimental Models: Attention Mechanism Involving all Facial Landmarks and Eyes Region

The separation between ’glare’ and ’normal’ is also not clean in the first model as seen from the plot, but is very distinct separation in the second. This clearly infers the impact of eye only attention that boosts up the focus on the eye region for the classification.

4.3. Final Model Comparison with Selected Literatures

Table 2. Experimental and Literature Comparison of the Final Model

Method/Source	RAF-DB	EER Dataset
MRE-CNN /ICANN	76.73%	
RC-DLP /CVPR	84.70%	
Emotion classifier /ICMI	80.00%	
DLP-CNN /TIP	84.13%	
IFSL (k-NN) /SP	72.60%	
IFSL (SVM) /SP	76.90%	
Capsnet /arXiv (2017)	76.12%	69.32%
Stacked-CNN /mdpi(2023)		77%
EyeAttention-Capsnet /My Experiment		82.27%

Comparing with the traditional methods in table 2, our final model outperforms many of the methods and is to the mark with the state-of-the-art methods of expression detection. Also, besides the facial expression domain, the model is compared with the work of (Mutanu 2023) in the terms of detecting eye expression. The mentioned final model for the task in the paper has the final accuracy of 77% in the dataset made by the authors. Our model has the accuracy of 82.27% in the similar and refined EER dataset. This presents the strong base for the inference that the method in our study is better in performance to (Mutanu 2023). This improvement has been achieved focusing on the accuracy performance of the model while keeping in mind the computational power of the modern digital devices. The model in (Mutanu 2023) is simpler compared to the model in this study but the model complexity has been compromised to enhance the recognition performance of the system which may be critical in various scenarios.

5. Conclusion

From the research work, the most important inference that standout considering the results is the application of Capsule Network in classification significantly improves the performance if the features fed to the capsule network are more pronounced. This research concludes the final classification model developed outperforms several traditional models and is considerably to the mark with some strong models, which shows the viability of the research. The final model has the accuracy of 82.27% which is a significant achievement considering the availability of the dataset targeted to recognize eye strain from the expression. Use of model in real time application is also feasible since the prediction time per single image of the model is considerably low once the model is loaded in the application.

However, there are some areas that can be improved in this domain such as diversification and expansion of dataset for eye strain expressions. Dynamic routing algorithm can be investigated for the possible modifications for the enhancement in performance.

References

[1] A. Barman, P. Dutta. "Influence of shape and texture features on FER." *IET Image Processing*, 2019: 1349-1363.

[2] "Boosting–poof: Boosting part based one vs one feature for facial expression recognition." *IEEE International Conference on Automatic Face & Gesture Recognition*, 2017: 967-972.

[3] C. Shan, S. Gong, P. McOwan. "Facial Expression Recognition based on local binary patterns: A comprehensive study." *Image and Vision Computing*, 2009: 803-816.

[4] D. Marrero, A. Guerrero, Tea. Ren. "Feratt: Facial expression recognition with attention net." *arXiv: 1902.03284*, 2019.

[5] D. Tran, L. Bourdev, Rea. Fergus. "Learning spatiotemporal features with 3d convolutional networks." *IEEE Conference on Computer Vision and Pattern Recognition.* Santiago,Chile, 2015. 3476-3483.

[6] K. Zhang, Y. Huang, Yea Du. "Facial expression recognition based on deep evolutional spatial-temporal networks." *Adaptive Behavior*, 2017: 4193-4203.

[7] L. O'Hare, T. Zhang, H. Nefs, P. Hibbard. "Visual Discomfort and Depth of field." *Iperception*, 2013: 156-169.

[8] Mutanu, L., Gohil, J., & Gupta, K. "Vision-autocorrect: A self-adapting approach towards relieving eye-strain using facial-expression recognition." *Software*, 2023: 197-217.

[9] S. Sabour, N. Frosst, and G. E. Hinton. "Dynamic routing between capsules." *Advances in neural information processing systems*, 2017: 30.

[10] Y. Fan, C. Lam, O. Victor. "Multiple-region ensemble convolutional neural network for facial expression recognition." *International conference on ANN*, 2018: 84-94.

[11] Y. Peng, H. Yin. "Apprgan: Appearance-based GAN for facial expression synthesis." *IET Image Processing*, 2019: 2706-2715.

[12] Y. Sun, X. Wang, X. Tang. "Deep CNN cascade for facial point detection." *IEEE Conference on Computer Vision and Pattern Recognition*, 2013: 3476-3483.

[13] Y. Yan, Z. Zhang, Sea. Chen. "Low-resolution facial expression recognition: A filter learning perspective." *Signal Processing*, 2020: 11-20.

[14] Y.Hu, Z.Zeng, L. Yin. "Multi-view facial expression recognition." *IEEE International Conference on Automatic Face and Gesture Recognition*, 2008: 1-6.

[15] Z. Liu, S. Li, W. Deng. "Real-world facial expression recognition using metric learning method." *Biometric Recognition*, 2016: 519-527.

[16] ZT. Liu, SH. Li, WH. Cao, DY. Li, M. Hao. "Combining 2d gabor and local binary pattern for FER using extreme learning machine." *Journal of Advanced Computational Intelligence and Intelligent Informatics*, 2019: 444-455.